

[0078] FIG. 2 is a multipoint processing method 100, in accordance with one embodiment of the present invention. The multipoint processing method 100 may for example be performed in the system shown in FIG. 1. The multipoint processing method 100 generally begins at block 102 where images are read from a multipoint input device, and more particularly a multipoint touch screen. By way of example, the multipoint touch screen may generally correspond to the multipoint touch screen disclosed in copending U.S. patent application Ser. No. 10/840,862, which is hereby incorporated herein by reference. Although the term “image” is used it should be noted that the data may come in other forms. In most cases, the image read from the touch screen provides magnitude (Z) as a function of position (x and y) for each sensing point or pixel of the touch screen. The magnitude may, for example, be reflect the capacitance measured at each point.

[0079] Following block 102, multipoint processing method 100 proceeds to block 104 where the image is converted into a collection or list of features. Each feature represents a distinct input such as a touch. In most cases, each feature includes its own unique identifier (ID), x coordinate, y coordinate, Z magnitude, angle .theta., area A, and the like. By way of example, FIGS. 3A and 3B illustrate a particular image 120 in time. In image 120, there are two features 122 based on two distinct touches. The touches may for example be formed from a pair of fingers touching the touch screen. As shown, each feature 122 includes unique identifier (ID), x coordinate, y coordinate, Z magnitude, angle .theta., and area A. More particularly, the first feature 122A is represented by ID.sub.1, x.sub.1, y.sub.1, Z.sub.1, .theta.sub.1, A.sub.1 and the second feature 122B is represented by ID.sub.2, x.sub.2, y.sub.2, Z.sub.2, .theta.sub.2, A.sub.2. This data may be outputted for example using a multitouch protocol.

[0080] The conversion from data or images to features may be accomplished using methods described in copending U.S. patent application Ser. No. 10/840,862 which is hereby incorporated herein by reference. As disclosed therein, the raw data is received. The raw data is typically in a digitized form, and includes values for each node of the touch screen. The values may be between 0 and 256 where 0 equates to no touch pressure and 256 equates to full touch pressure. Thereafter, the raw data is filtered to reduce noise. Once filtered, gradient data, which indicates the topology of each group of connected points, is generated. Thereafter, the boundaries for touch regions are calculated based on the gradient data, i.e., a determination is made as to which points are grouped together to form each touch region. By way of example, a watershed algorithm may be used. Once the boundaries are determined, the data for each of the touch regions are calculated (e.g., x, y, Z, .theta., A).

[0081] Following block 104, multipoint processing method 100 proceeds to block 106 where feature classification and groupings are performed. During classification, the identity of each of the features is determined. For example, the features may be classified as a particular finger, thumb, palm or other object. Once classified, the features may be grouped. The manner in which the groups are formed can widely varied. In most cases, the features are grouped based on some criteria (e.g., they carry a similar attribute). For example, the two features shown in FIGS. 3A and 3B may be grouped together because each of these features is located in proximity to each other or because they are from the same hand. The grouping may include some level of filtering to filter out

features that are not part of the touch event. In filtering, one or more features may be rejected because they either meet some predefined criteria or because they do not meet some criteria. By way of example, one of the features may be classified as a thumb located at the edge of a tablet PC. Because the thumb is being used to hold the device rather than being used to perform a task, the feature generated therefrom is rejected, i.e., is not considered part of the touch event being processed.

[0082] Following block 106, the multipoint processing method 100 proceeds to block 108 where key parameters for the feature groups are calculated. The key parameters may include distance between features, x/y centroid of all features, feature rotation, total pressure of the group (e.g., pressure at centroid), and the like. As shown in FIG. 4, the calculation may include finding the centroid C, drawing a virtual line 130 to each feature from the centroid C, defining the distance D for each virtual line (D.sub.1 and D.sub.2), and then averaging the distances D.sub.1 and D.sub.2. Once the parameters are calculated, the parameter values are reported. The parameter values are typically reported with a group identifier (GID) and number of features within each group (in this case three). In most cases, both initial and current parameter values are reported. The initial parameter values may be based on set down, i.e., when the user sets their fingers on the touch screen, and the current values may be based on any point within a stroke occurring after set down. As should be appreciated, blocks 102-108 are repetitively performed during a user stroke thereby generating a plurality of sequentially configured signals. The initial and current parameters can be compared in later steps to perform actions in the system.

[0083] Following block 108, the process flow proceeds to block 110 where the group is or associated to a user interface (UI) element. UI elements are buttons boxes, lists, sliders, wheels, knobs, etc. Each UI element represents a component or control of the user interface. The application behind the UI element(s) has access to the parameter data calculated in block 108. In one implementation, the application ranks the relevance of the touch data to the UI element corresponding there to. The ranking may be based on some predetermine criteria. The ranking may include producing a figure of merit, and whichever UI element has the highest figure of merit, giving it sole access to the group. There may even be some degree of hysteresis as well (once one of the UI elements claims control of that group, the group sticks with the UI element until another UI element has a much higher ranking). By way of example, the ranking may include determining proximity of the centroid (or features) to the GUI object associated with the UI element.

[0084] Following block 110, the multipoint processing method 100 proceeds to blocks 112 and 114. The blocks 112 and 114 can be performed approximately at the same time. From the user perspective, in one embodiment, the blocks 112 and 114 appear to be performed concurrently. In block 112, one or more actions are performed based on differences between initial and current parameter values as well as the UI element to which they are associated. In block 114, user feedback pertaining to the one or more action being performed is provided. By way of example, user feedback may include display, audio, tactile feedback and/or the like.

[0085] FIG. 5 is a parameter calculation method 150, in accordance with one embodiment of the present invention. The parameter calculation method 150 may, for example, correspond to block 108 shown in FIG. 2. The parameter calculation method 150 generally begins at block 152 where